

Tarea 9

Axel Báez Ochoa
Claudia Acosta Díaz

18 de octubre de 2020

1. Problema 4

Primero veamos que para arboles rojos-negros las restricciones que debe seguir son las siguientes:

- 1.- Cada nodo tiene un elemento (un label) llamado color, que es o rojo, o negro.
- 2.- La raíz del ABB es siempre negra.
- 3.- Los hijos de un nodo rojo tienen que ser ambos negros (o sea, no puede haber two rojos consecutivos en un camino de la raíz a una hoja).
- 4.- Las ligas vacías (apuntadores NULL en nodos terminales) cuentan como negro.
- 5.- Cualquier camino de un nodo v del árbol hacia un NULL tiene el mismo numero de nodos negros (sin contar v). Ese numero se llama altura negra de v y lo notaremos $hb(v)$.

Comenzamos con un árbol rojo-negro sin violaciones, y agregamos un nuevo nodo coloreado como rojo. Si ese nodo es la raíz cambiamos su color a negro por lo que la propiedad (2) no sera violada, la propiedad (1) se mantiene porque el nuevo nodo va a ser de color rojo, o negro si es la raíz. La Propiedad (5) que dice que el número de nodos negros es el mismo en cada ruta desde un nodo dado, también está satisfecho, ya que antes de insertar este nuevo nodo, el árbol cumplía esta restricción por lo que el nuevo nodo reemplaza un nodo negro NULL, y como este nuevo nodo es rojo con hijos NULL ,estos cuentan como negros según la propiedad (4) por lo que el numero de nodos negros no se vera afectado. Por lo que la única propiedad que puede fallar es la numero (3) esto se debe a que si el nuevo nodo no es la raíz su color sera rojo, por lo que si el padre de este es rojo se violara la restricción.

2. Problema 5

Teníamos un árbol que cumplía las restricciones , hasta que se inserto un nuevo nodo que viola la propiedad (3), ya que el padre de este es rojo, si el tío de este es rojo, como el árbol cumplía las propiedades antes de la inserción del nuevo nodo, el abuelo debe de ser negro , por lo que podemos cambiar los colores del papa y el tío del nuevo nodo a negros, solucionando el problema

de que el nuevo nodo y su papa eran rojos, y cambiamos el color del abuelo a rojo para mantener la propiedad (5), ahora podría ocurrir que el papa del abuelo sea rojo violando nuevamente la restricción por lo que se hará el mismo proceso pero ahora tomando al abuelo, donde el padre de este no cambiara de color, si el abuela es la raíz se violara la restricción (2), si no es la raíz se corrigió el problema y no se ah violado (2). Esto se repetirá subiendo en el arbol por lo que entonces sera $O(\log n)$. Todo este proceso cambiara el color de unos nodos en cada iteración pero preserva la propiedad (5) done todo camino de un nodo hacia un NULL tienen el mismo número de negros.

3. Problema 6

Teníamos un árbol que cumplía las restricciones , hasta que se inserto un nuevo nodo que viola la propiedad (3), ya que el padre de este es rojo, si el tío de este es negro, si el nuevo nodo insertado es un hijo del lado derecho de su padre, usamos una rotación a la izquierda, para que se vuelva un hijo izquierdo, como este nuevo nodo y su papa son rojos, la rotación no afecta el numero de nodos negros para los caminos, propiedad (5), el nodo abuelo seguirá siendo el mismo , dada la forma en que esta definida la rotación, hacemos una rotación a la derecha y cambios de colores y preservamos (5) y ya que no tenemos dos nodos rojos, uno papa de otro, terminamos. Si el hijo es izquierdo este apuntara al padre que es rojo, si es derecho el padre sera negro, así que si es la raíz se solucionara el problema de el inciso 5 donde volvíamos la raíz roja. Como la altura del árbol de n nodos es $O(\log n)$, le tomara a esto $O(\log n)$, como ya dijimos este paso solo se repite una vez pero si sucede lo que mencionamos en el Problema 5, como mencionamos se podría repetir $O(\log n)$, así que la complejidad de arreglar el árbol sigue siendo $O(\log n)$, donde nunca hay mas de 2 rotaciones pues las restricciones se solucionan si entramos a lo descrito en este problema y solo continúan cuando estamos en lo descrito para el problema 5.